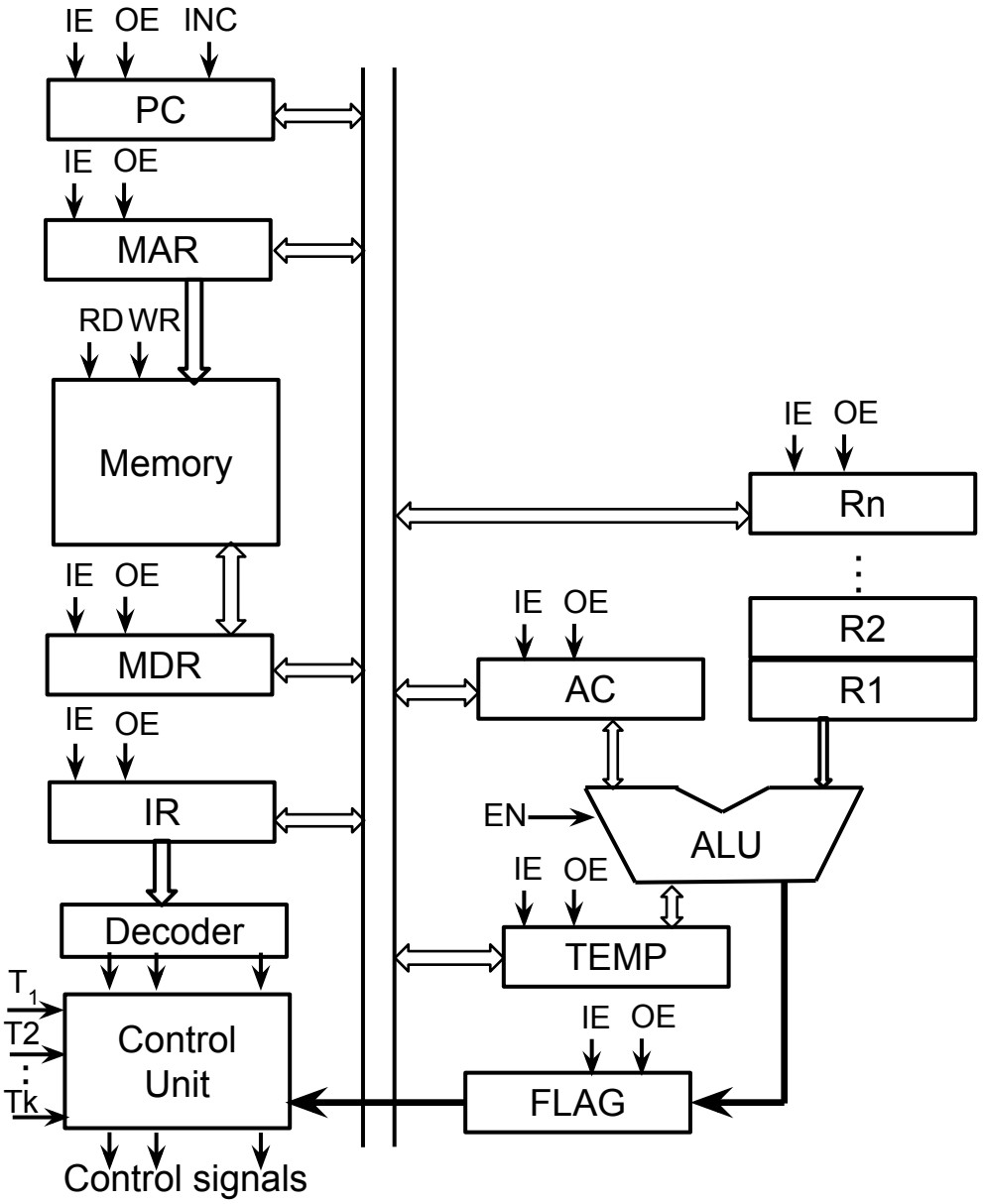
Data path design



Instruction Fetch

Fetch the instruction form memory to IR

Sequence of micro operations of fetch cycle

T1: PC -> MAR OE_{PC} , IE_{MAR}

T2: M(MAR) -> MDR RD , $IE_{MDR,}$ OE_{MAR}

T3: MDR -> IR OE_{MDR} , IE_{IR}

T4: PC <- PC + 1 INC_{PC}

ADD R1 AC = AC + R1

T1: TEMP <- AC + R1 OE_{AC}, $OE_{R1,}$ IE_{TEMP} , EN

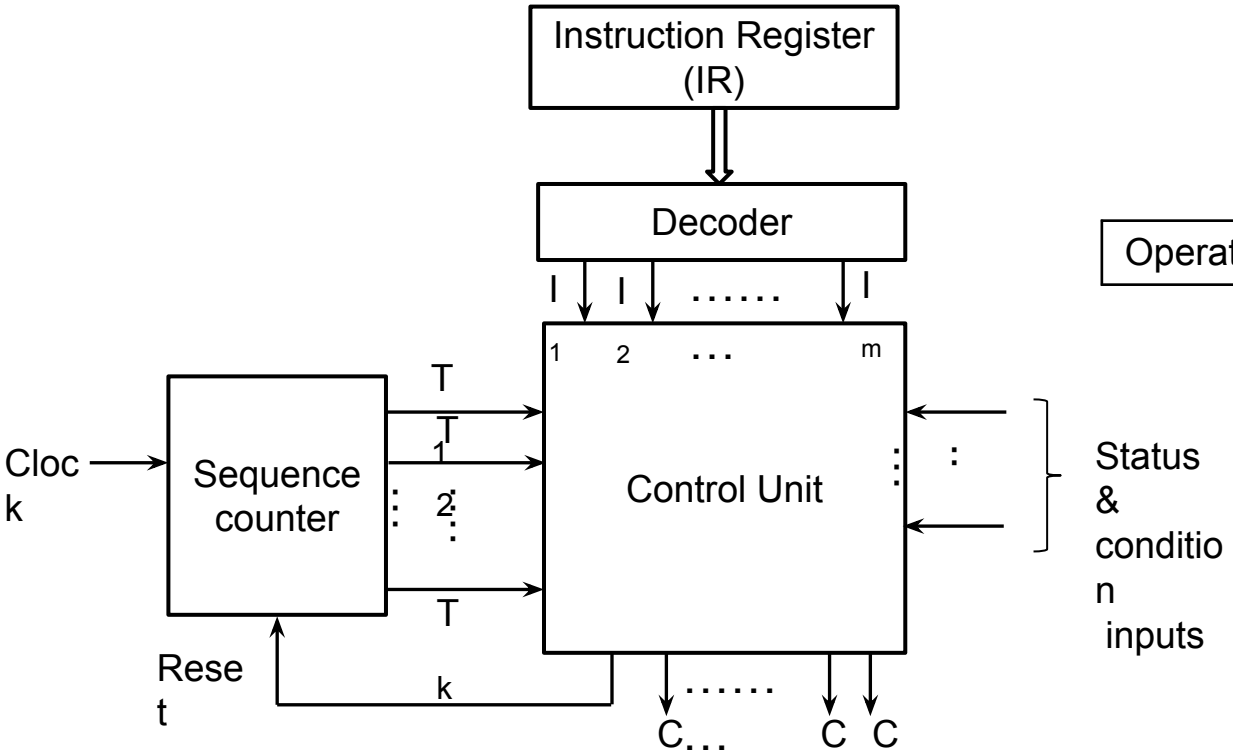
T2: AC <- TEMP IE_{AC} , OE_{TEMP}

Instruction Cycle

Time required to complete the execution of an instruction is called Instruction Cycle.
The operations of instruction cycle are given below.
IF (Instruction Fetch)
ID (Instruction Decoding)
OF(Operand Fetch)
EX (Execution)
WB / ST (Write Back / Store)

- **Instruction Fetch:** Fetch the instruction form memory to IR (Instruction Register).
- **Instruction Decoding:** Instruction is decoded and tells the processor what has to be done?
- **Operand fetch:** Fetch the operand from memory if it is required.
- **Execution:** Actual operations (processor operations) are performed.
- **Write Back:** The result will be stored back to the memory location if it is required.

Hardwired control unit

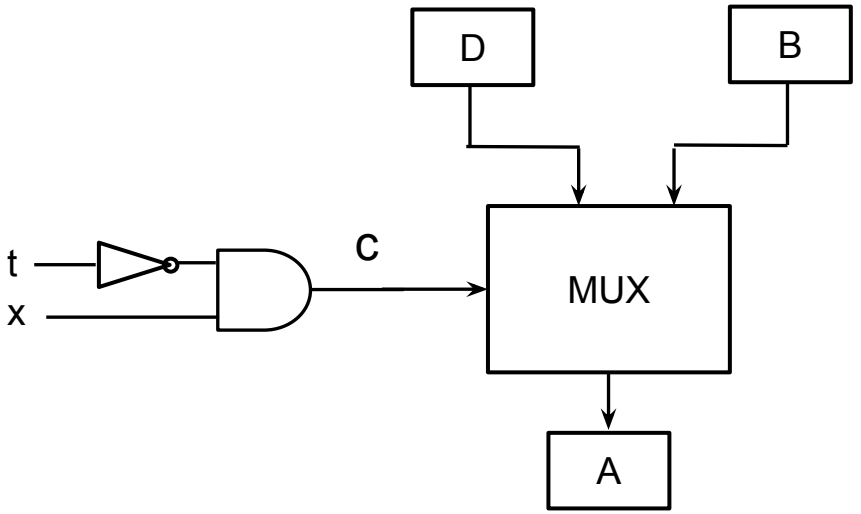


Operation Code	Mode	Address

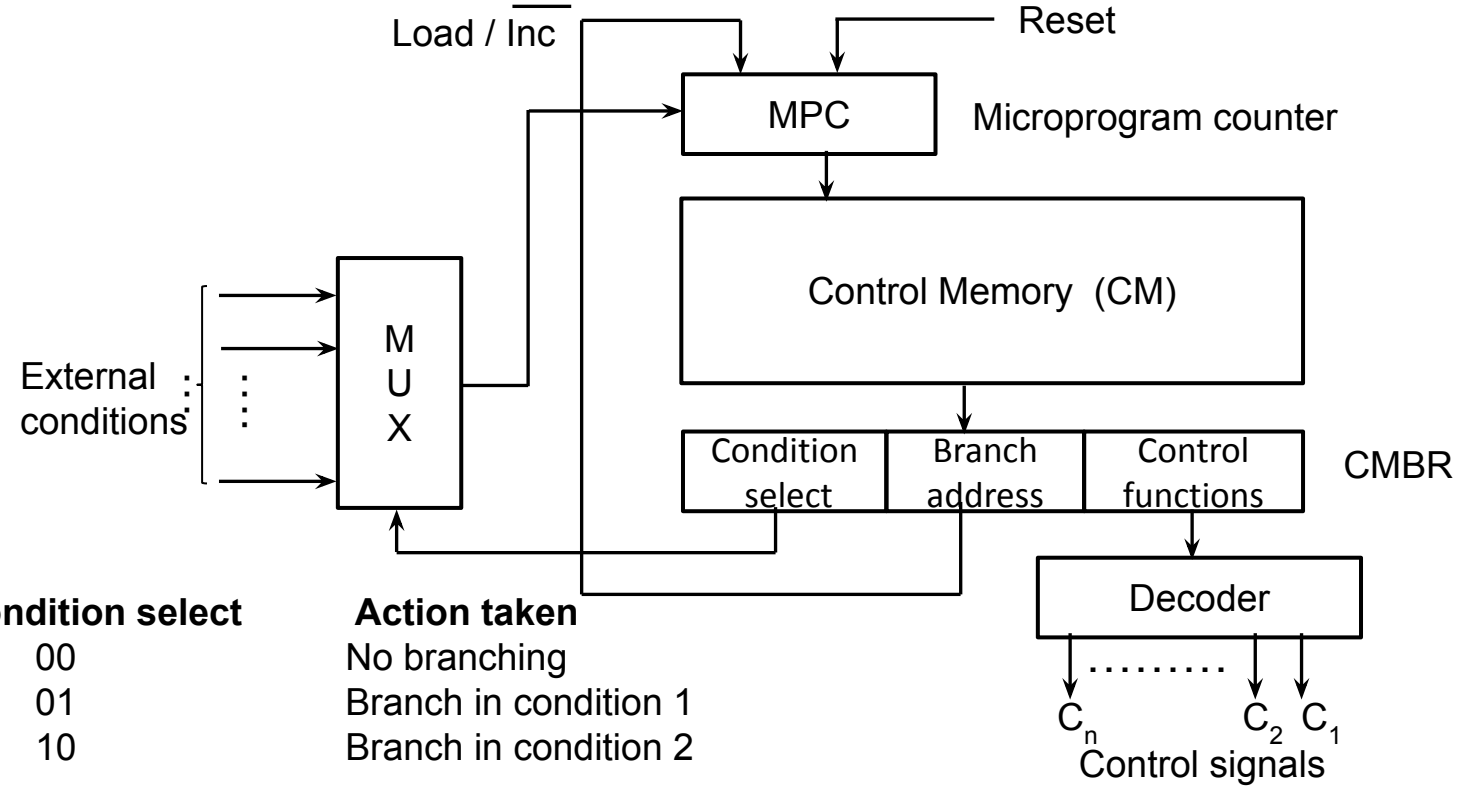
- The instruction that is loaded in the IR is decoded by the instruction decoder.
- If the IR is an 8-bit register, then the instruction decoder generates 2^8 (256) lines. That means 256 individual instructions are there.
- The number of signals generated by the control signal generator combines with the timing signals.
- Depending upon status and condition inputs final control signals ($C_1, C_2, \ldots C_n$) are generated

Hardware implementation of a function

If t = 0 and x =1
then A <- B
else A <- D



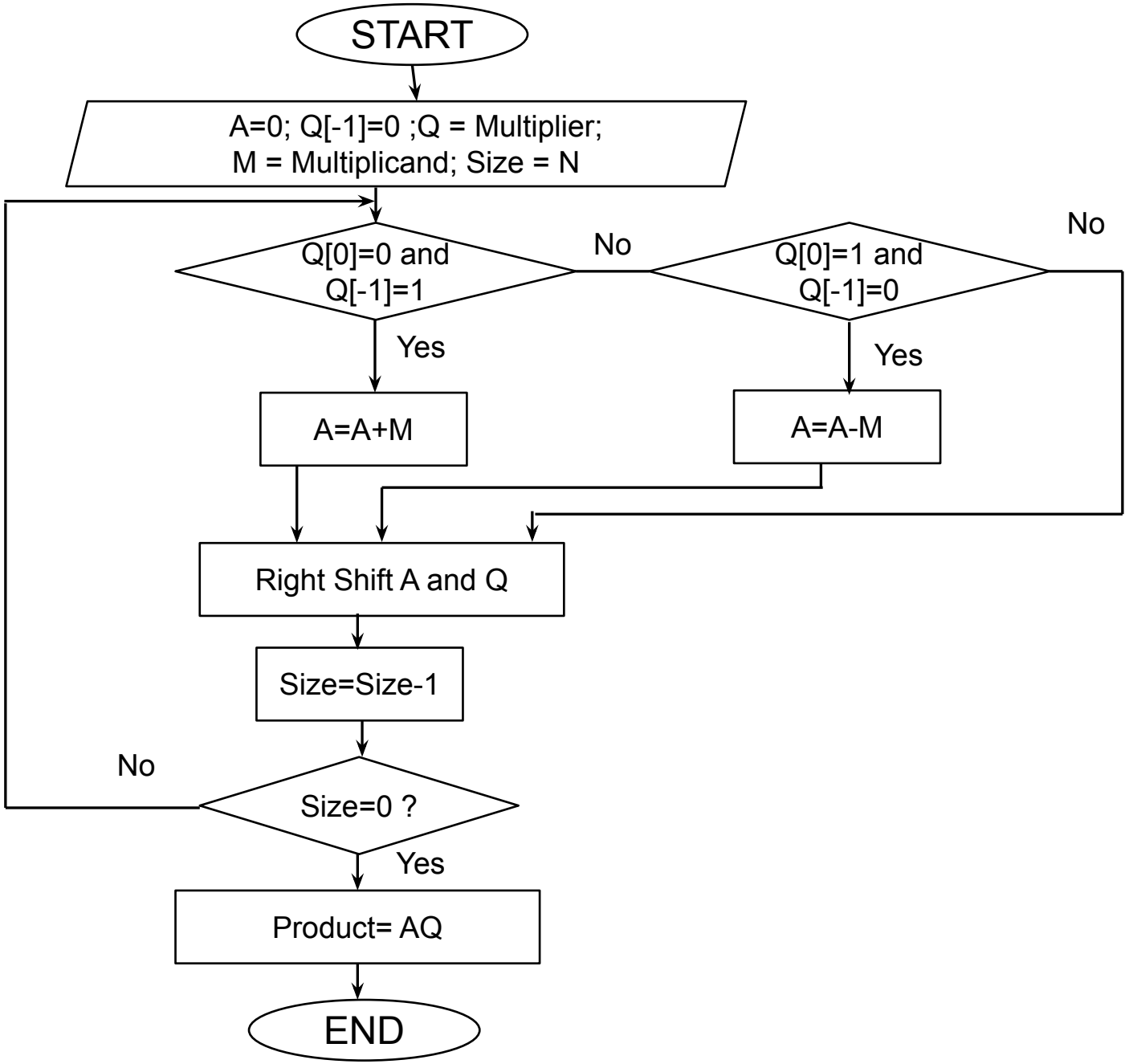
General Purpose microprogrammed control unit



Condition select	Action taken
00	No branching
01	Branch in condition 1
10	Branch in condition 2
-	--------

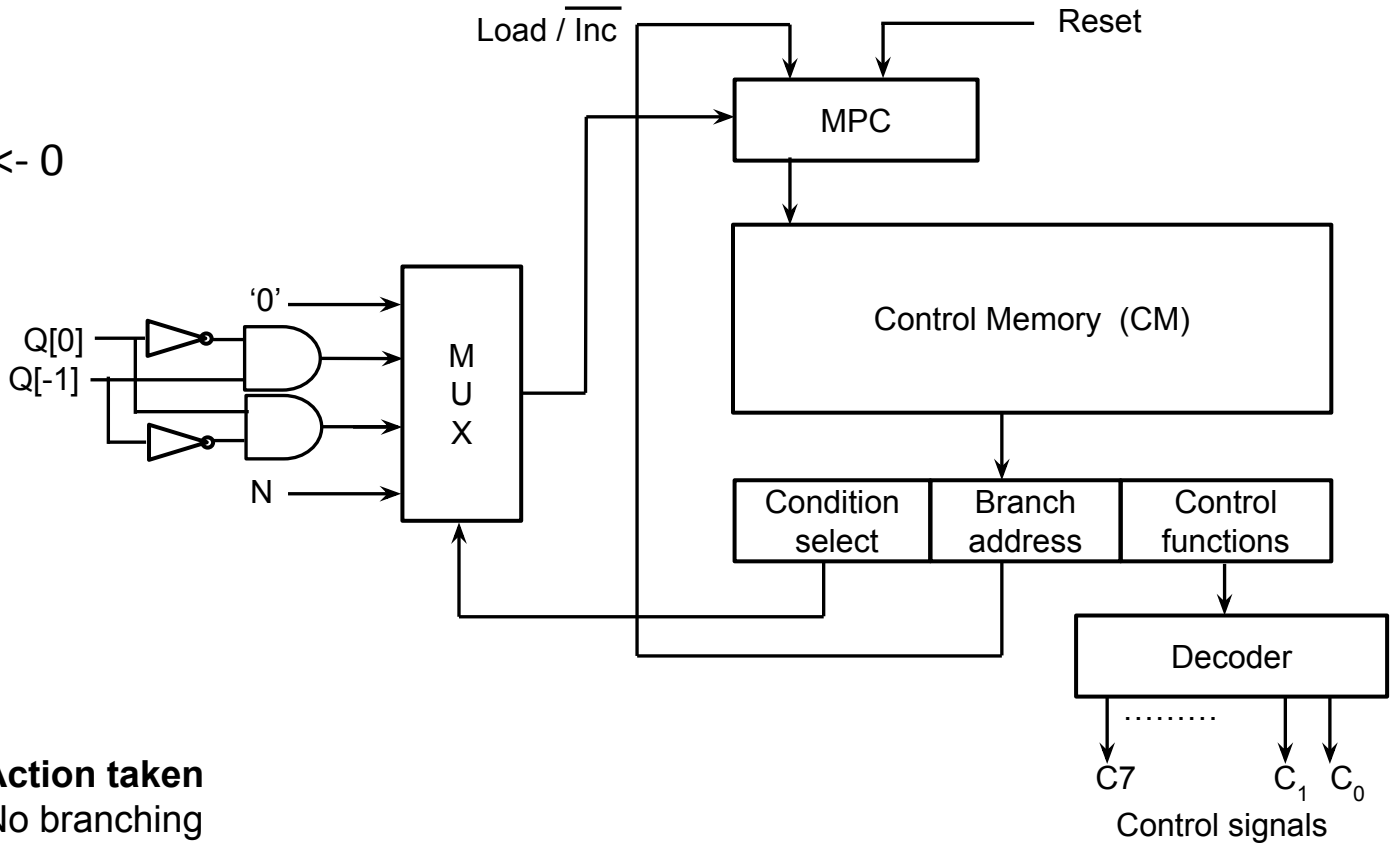
- The micro instructions are kept in control memory which is pointed by microprogram counter.
- The Control Memory Buffer Register (CMBR) holds the microinstruction temporary. It is divided into three field condition select, branch address and control function.
- If there is a branching then load the MPC with this branch address otherwise increment the MPC.
- Control function generates the control signals (C_1 ,C_2 ,.....C_n) which are separated by decoder.

Booths Multiplication Flow chart



Booth's Multiplier control unit (Microprogrammed) design

C0: AC <- 0
C1: M <- Input
C2: N <- 4
C3: Q[3:0]<- Input, Q[-1] <- 0
C4: AC <- AC+M
C4': AC <- AC-M
C5: ASR(AC & Q)
C6:N <- N-1
C7: Output =AC & Q



Condition select	**Action taken**
00	No branching
01	Branch in condition 1 (Q[0]=0 and Q[-1]=1)
10	Branch in condition 2 (Q[0]=1 and Q[-1]=0)
11	Branch if N is nonzero

Difference between Microprogrammed and Hardwired Approach

Hardwired control unit generates the control signals implementing the logic circuits.
Microprogrammed control unit generates the control signals with the help of micro instructions stored in control memory.

Hardwired control unit based on hardware which is faster compared to microprogrammed control unit.
Microprogrammed control unit is software based which is slower because of micro instructions are used for generating signals.

Hardwired control unit designing is difficult to modify because of changing the hardware circuit.
Microprogrammed is easy to modify as the modification is based on instruction level.

In hardwired control, it cannot handle complex instructions as the circuit design becomes complex.
In microprogrammed control, it can handle complex instructions.

Hardwired control unit used in Reduced Instruction Set Computers(RISC).Some olden day computers used a hardwired control unit approach.
Microprogrammed control unit used in Complex Instruction Set Computers(CISC). Now a days microprogramming is accepted as a standard tool to design the control unit of a computer.

Thank You